REMARKS

The Office Action dated March 23, 2005 has been received and carefully noted. The following remarks are submitted as a full and complete response thereto. Claims 1-71 are currently pending in the application. The Office Action indicated that claims 38-60 have been allowed. Therefore, claims 1-37 and 61-71 are respectfully submitted for consideration.

In the Office Action, claims 8, 9, 15, 69, and 70 were rejected under 35 U.S.C. §102(e) as being anticipated by Larson (U.S. Patent No. 6,115,705). The rejection is respectfully traversed for the reasons which follow.

Claim 8, upon which claims 9-15 are dependent, recites a method including the steps of rotating entries stored in a plurality of locations of a hash bucket to empty a first location, and adding a key to the first location of the plurality of locations of the hash bucket.

Claim 69, upon which claims 70 and 71 are dependent, recites a machine-readable medium which provides instructions. When those instructions are executed by a machine, they cause the machine to perform the operations of rotating entries stored in a plurality of locations of a hash bucket to empty a first location, and adding a key to the first location of the plurality of locations of the hash bucket.

As will be discussed below, Larson fails to disclose or suggest all of the elements of claims 8 and 69, and therefore fails to provide the features discussed above.

Larson discloses a relational database system and method for query processing using early aggregation. More specifically, Larson provides a relational database system with a non-volatile memory, a volatile memory for temporarily storing a set of data records, and a query processor. The volatile memory has an amount of available space for query processing that is segmented into multiple memory pages. The memory pages are initially empty and available in a pool for use by the query processor. The query processor establishes a partition table that defines multiple partitions. The query processor hashes incoming data records to an entry in the partition table. The entry directs the data records to the appropriate partitions. One memory page is initially assigned to each partition or alternatively taken from the free pool as needed by the partition. The data records are stored in the memory pages associated with the partitions. Before a new data record is placed into a particular partition, the query processor attempts to aggregate the new data record with any like data record that already exists in the particular partition.

Applicants respectfully submit that Larson fails to disclose or suggest "rotating entries stored in a plurality of locations of a hash bucket to empty a first location," as recited in claims 8 and 69. According to the present invention, if not all the locations in the hash bucket are occupied, the CPU 3 may direct a reordering unit 28 to reorder the entries by inserting the new key entry into the first location in the bucket. For example, if there were valid entries present in location 1 and location 2 prior to the addition of the new key entry, then at completion of the reordering operation by the reordering unit 28,

the new entry will occupy location 1 and the entry that used to occupy location 1 will move to location 2, and so on (Specification, paragraph 39 and figure 3).

Larson, on the other hand, does not disclose or suggest this aspect of the claimed invention. According to Larson, as a new data record is placed into a particular partition, a query processor attempts to aggregate the new data record with any like data record that already exists in that particular partition (Larson, Column 2, lines 51-56, Column 6, lines 1-65, and Figure 4). Furthermore, according to Larson, an overflow condition is reached when the query processing program attempts to add a new data record to a full memory page. In this situation, Larson discloses that the relational database program appends a memory page from the free list to the appropriate partition and links the memory page to the existing chain for that partition (Larson, Column 7, lines 13-19). Larson, however, does not disclose or suggest rotating entries stored in a plurality of locations of a hash bucket to empty a first location, as recited in claims 8 and 69. Therefore, for at least the reasons discussed above, Applicants respectfully request that the rejection of claims 8 and 69 be withdrawn.

Claims 9, 15, and 70 are dependent upon claims 8 and 69, respectively. As such, Applicants submit that claims 9, 15, and 70 should also be allowed for at least their dependence upon claims 8 and 69, and for the specific limitations recited therein.

Claims 16 and 19 were rejected under 35 U.S.C. §102(b) as being anticipated by Nemes (U.S. Patent No. 5,893,120). The rejection is respectfully traversed for the following reasons.

Claim 16, upon which claims 17-19 are dependent, recites a method including the steps of receiving a unique key to be deleted, and searching a plurality of locations of a hash bucket for a match to the unique key. The method further includes the steps of deleting the unique key from a location of the plurality of locations in the hash bucket upon finding the match to the unique key in the location, and rotating remaining entries of the hash bucket after the deleting the unique key from the location of the plurality of locations in the hash bucket.

As will be discussed below, Nemes fails to disclose or suggest all of the elements of claims 16 and 19, and therefore fails to provide the features discussed above.

Nemes discloses a method and apparatus for performing storage and retrieval in an information storage system using a hashing technique with an external chaining method for collision resolution. A garbage collection technique is used to remove all expired records stored in the system in the external chain targeted by a probe into the data storage system. During normal data insertion or retrieval probes into the data store, the entire linked-list chain of records is searched for expired items which are then removed.

Applicants respectfully submit that Nemes fails to disclose or suggest rotating remaining entries of the hash bucket after deleting the unique key from the location of the plurality of locations in the hash bucket, as recited in claim 16. Nemes only discloses that the deletion of records involves only adjusting the pointers to bypass the deleted record and retruning the storage it occupied to the available storage pool (Nemes, Comumn 5, lines 31-33 and Column 6, lines 21-34). As such, Nemes does not disclose

rotating the entries after deleting the unique key, as recited in present claim 16. Thus, Nemes fails to disclose or suggest all of the elements of claim 16 and Applicants respectfully request that the rejection of claim 16 be withdrawn.

Claim 19 is dependent upon claim 16. Consequently, claim 19 should also be allowed for at least its dependence upon claim 16, and for the specific limitations recited therein.

Claims 20 and 21 were rejected under 35 U.S.C. §102(b) as being anticipated by Kass (U.S. Patent No. 5,566,324). The rejection is respectfully traversed for the reasons which follow.

Claim 20, upon which claims 21 and 22 are dependent, recites a method including the steps of associating an entry of a plurality of entries in a cache with a timestamp, incrementing the timestamp, and deleting the entry of the plurality of entries in the cache based on a value of the timestamp.

As will be discussed below, Kass fails to disclose or suggest all of the elements of claims 20 and 21, and therefore fails to provide the features discussed above.

Kass discloses a method and computer system including a main memory prefetch cache. The computer system includes a processor and a processor cache coupled to the processor. The computer system further includes a memory controller coupled to the processor and a main memory coupled to the memory controller. The memory controller includes a main memory prefetch cache and a cache control circuit. The cache control circuit is coupled to the prefetch cache and is used to determine whether a cache hit has

occurred, where the current line requested by the processor is stored in the prefetch cache. If a cache hit has occured, then the control circuit causes the retrieving of the current line from the prefetch cache for use by the processor and also causes the overwriting of the current line in the prefetch memory with the next line from the main memory. If the prefetch cache does not contain the current line requested by the processor, thereby signifying a cache miss, then current line is retrieved from the main memory for use by the processor. The next line is then retrieved from the main memory and is stored in the prefetch cache.

Applicants respectfully submit that Kass does not disclose or suggest a timestamp, as recited in claims 20 and 21. Rather, Kass only discloses the use of a counter which contains a count value indicating how recently the contents of the prefetch cache register have been used (Kass, Column 4, lines 51-56). This does not correspond to the timestamp of the present invention. According to the claimed invention, a timestamp is provided which may be incremented every second, and wrapped around to 0 when the value of the timestamp reaches a predetermined maximum value (Specification, paragraph 0050). Kass does not disclose or suggest such a timestamp. Therefore, Kass fails to disclose or suggest all of the elements of claims 20 and 21. As such, Applicants respectfully request that the rejection of claims 20 and 21 be withdrawn.

Additionally, claim 21 is dependent upon claim 20. Therefore, claim 21 should also be allowed for at least its dependence upon claim 20, and for the specific limitations recited therein.

Claims 20 and 22 were rejected under 35 U.S.C. §102(e) as being anticipated by Bogin (U.S. Patent No. 6,658,533). The rejection is respectfully traversed for the reasons which follow.

Claim 20, upon which claims 21 and 22 are dependent, recites a method including the steps of associating an entry of a plurality of entries in a cache with a timestamp, incrementing the timestamp, and deleting the entry of the plurality of entries in the cache based on a value of the timestamp.

As will be discussed below, Bogin fails to disclose or suggest all of the elements of claims 20 and 22, and therefore fails to provide the features discussed above.

Bogin discloses a write cache that reduces the number of memory accesses required to write data to main memory. When a memory write request is executed, the request updates the relevant location in cache memory and also updates the corresponding location in main memory. A separate write cache is dedicated to temporarily holding multiple write requests so that they can be organized for more efficient transmission to memory in burst transfers. All writes within a predefined range of addresses can be written to memory as a group. Entries may be held in the write cache until a minimum number of entries are available for writing to memory, and a least-recently-used mechanism can be used to decide which entries to transmit first.

Bogin, like Kass, does not disclose or suggest the timestamp of the present invention, as recited in claims 20 and 22. Rather, Bogin discloses assigning a counter value to each entry as it is placed in cache, and incrementing the counter after each

assignment (Bogin, Column 5, lines 46-50). Bogin does not disclose or suggest a timestamp which may be incremented every second, and wrapped around to 0 when the value of the timestamp reaches a predetermined maximum value (Specification, paragraph 0050). Therefore, Bogin does not disclose the timestamp of the claimed invention. As such, Applicants respectfully request that the rejection of claims 20 and 22 over Bogin be withdrawn.

In addition, claim 22 is dependent upon claim 20, and therefore claim 22 should also be allowed for at least its dependence upon claim 20 and for the specific limitations recited therein.

Claim 23 was rejected under 35 U.S.C. §102(e) as being anticipated by Huang (U.S. Patent No. 6,683,887). The rejection is respectfully traversed for the following reasons.

Claim 23, upon which claims 24-37 are dependent, recites a method including the steps of identifying a subscriber associated with a particular packet utilizing a line card, and transmitting the particular packet to one of a plurality of data cards associated with the identified subscriber.

As will be discussed below, Huang fails to disclose or suggest all of the elements of claim 23, and therefore fails to provide the features discussed above.

Huang discloses an asymmetrical digital subscriber line (ADSL) downstream high speed cell bus interface protocol and a system for the downstream transmission of ADSL traffic. More specifically, Huang discloses synchronizing frame boundaries of an

asymmetrical digital subscriber line (ADSL) frame by using a subscriber bus interface (SBI) frame consisting of a predetermined number of SBI time slots. Huang further discloses assigning a plurality of cell packets to the ADSL frame, the number of cell packets in the ADSL frame different from the number of SBI time slots in the SBI frame, and providing an internal cell for transmission within one of the cell packets, the internal cell comprising a plurality of routing tag bytes and a plurality of payload bytes.

Applicants respectfully submit that Huang fails to disclose or suggest identifying a subscriber associated with a particular packet utilizing a line card, and transmitting the packet to a data card associated with the identified subscriber, as recited in claim 23. Huang makes no mention of identifying a subscriber or transmitting a packet to a data card associated with the identified subscriber. Huang merely discloses an ADSL line unit 88 which includes a router 90 for routing the data carried by an ADSL cell packet to downstream destination ports. Huang does not disclose identifying a subscriber or transmitting the packet to a data card associated with that identified subscriber. Consequently, Huang fails to disclose or suggest all of the elements of claim 23, and, as such, Applicants respectfully request that the rejection of claim 23 be withdrawn.

Claims 1-3, 61, 62, 65 and 66 were rejected under 35 U.S.C. §103(a) as being unpatentable over Spinney (U.S. Patent No. 5,390,173) in view of Handy "The Cache Memory Book." The Office Action took the position that Spinney discloses all of the elements of the claims, with the exception of the second structure used to perform the

search being a cache. The Office Action then relies upon Handy as allegedly curing this deficiency in Spinney. The rejection is respectfully traversed for the following reasons.

Claim 1, upon which claims 2-7 are dependent, recites a method which includes the steps of receiving a unique key, searching a hash for a match to the unique key, searching a cache for the match to the unique key concurrently with the searching the hash for the match to the unique key, and obtaining information regarding the unique key.

Claim 61, upon which claims 62-64 are dependent, recites an apparatus including means for receiving a unique key, means for searching a hash for a match to the unique key, means for searching a cache for the match to the unique key concurrently with searching the hash for the match to the unique key, and means for obtaining information regarding the unique key.

Claim 65, upon which claims 66-68 are dependent, recites a machine-readable medium that provides instructions. When those instructions are executed by a machine, they cause the machine to perform the operations of receiving a unique key, searching a hash for a match to the unique key, searching a cache for the match to the unique key concurrently with the searching the hash for the match to the unique key, and obtaining information regarding the unique key.

As will be discussed below, Spinney and Handy fail to disclose or suggest all of the elements of claims 1, 61, and 65, and therefore fails to provide the features discussed above.

Spinney discloses a method for source address and destination address lookups. The method searches a database using a combination of programmable hash algorithms, binary search algorithms, and a small content-addressable memory (CAM). Spinney teaches the use of two different techniques to perform the address lookup. Most of the lookups are obtained by hashing the address, using a programmable 48-bit to 48-bit linear hash function, and then using the low-order 16-bits, to obtain one of 64K hash buckets, where each hash bucket contains both a pointer to a set of zero to seven lookup records, and the number of lookup records in this set. The second technique is used to handle the situation where more than seven 48-bit addresses hash to the same bucket. To handle this situation, one of the addresses is simply placed in a CAM memory chip. The result of the comparison is either (a) tile entry is not found in the CAM, or (b) a number (0-to-255) supplying the index of tile associated lookup record in a translation table.

Applicants respectfully submit that Spinney and Handy, whether considered alone or in combination, fail to disclose or suggest concurrently searching both the cache and the hash for the match to the unique key, as recited in present claims 1, 61, and 65. Although Spinney discloses the use of two different techniques in performing address lookups, it does not disclose searching the cache for the match to the unique key concurrently with searching the hash for the match to the unique key, as recited in present claims 1, 61, and 65. Additionally, Handy also fails to cure this deficiency in Spinney. Consequently, the combination of Spinney and Handy fails to disclose or suggest at least concurrently searching both the cache and the hash for the match to the unique key. As

such, Applicants respectfully request that the rejection of claims 1, 61, and 65 be withdrawn.

Claims 2, 3, 62, and 66 are dependent upon claims 1, 61, and 65, respectively. Therefore, claims 2, 3, 62, and 66 should also be allowed for at least their dependence upon claims 1, 61, and 65, and for the specific limitations recited therein.

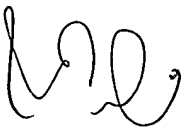
Claims 4-7, 10-14, 17, 18, 24-37, 63, 64, 67, 68, and 71 were objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims. Applicants note that claims 4-7, 10-14, 17, 18, 24-37, 63, 64, 67, 68, and 71 are dependent upon claims 1, 8, 16, 23, 61, 65, and 69, respectively. As discussed above, Applicants respectfully assert that claims 1, 8, 16, 23, 61, 65, and 69 recite subject matter which is neither disclosed nor suggested by the cited prior art. Thus, claims 4-7, 10-14, 17, 18, 24-37, 63, 64, 67, 68, and 71 should be allowed for at least their dependence upon claims 1, 8, 16, 23, 61, 65, and 69, and for the specific limitations recited therein. As such, for at least the reasons discussed above, Applicants respectfully submit that these claims are allowable in their current form.

Applicants respectfully submit that the cited prior art fails to disclose or suggest critical and important elements of the claimed invention. These distinctions are more than sufficient to render the claimed invention unanticipated and unobvious. It is therefore respectfully requested that all of claims 1-71 be allowed, and this application passed to issue.

If for any reason the Examiner determines that the application is not now in condition for allowance, it is respectfully requested that the Examiner contact, by telephone, the applicants' undersigned attorney at the indicated telephone number to arrange for an interview to expedite the disposition of this application.

In the event this paper is not being timely filed, the applicants respectfully petition for an appropriate extension of time. Any fees for such an extension together with any additional fees may be charged to Counsel's Deposit Account 50-2222.

Respectfully submitted,



Majid S. AlBassam
Registration No. 54,749

Customer No. 32294
SQUIRE, SANDERS & DEMPSEY LLP
14TH Floor
8000 Towers Crescent Drive
Tysons Corner, Virginia 22182-2700
Telephone: 703-720-7800
Fax: 703-720-7802
MSA:jf